

```

1 < /* 25.03.21 2025년 3월 27일 목요일 오후 4:37
2 주의사항)
3 1. 키워드, 테이블이름, 컬럼이름 대/소문자 구별 X
4 2. 들여쓰기, 개행
5 3. 문장 종결 ;
6
7 데이터 조회(Select)
8 SELECT *| COLUMN NAME | EXPRESSION | AS ALIAS
9 FROM TABLE_NAME
10 ----OPTION
11 WHERESEARCH_CONDITION
12 GROUP BY
13 HAVING
14 ORDER BY ;
15
16 **/
17
18 SELECT *
19 FROM EMPLOYEE; /*employee에 전이*/
20
21
22 SELECT *
23 FROM DEPARTMENT;
24
25 SELECT *
26 FROM JOB j ;
27
28 SELECT *
29 FROM LOCATION;
30
31 SELECT *
32 FROM COUNTRY;
33
34 SELECT *
35 FROM SAL_GRADE sg ;
36
37 SELECT EMP_ID ,
38        EMP_NAME ,
39        SALARY*1.1 AS SALARY,
40        '재직' AS 근무여부 /*dummy column 생성가능*/
41 FROM EMPLOYEE;
42
43 --DISTINCT (중복 제거)
44 SELECT DISTINCT DEPT_ID/*distinct는 단일사용만 가능*/
45 FROM EMPLOYEE;
46
47 -- WHERE: 행 제한
48 SELECT *
49 FROM EMPLOYEE

```

```

50 WHERE SALARY >= 3000000 OR DEPT_ID = 60; /* 목시적 캐스팅 발생 |
연산자: AND | OR */
51
52 -- || : 연결연산자 (컬럼과 컬럼을 연결할 때)
53 /*
54 =, >=, <=, >, <, !=
55 BETWEEN ~ AND
56 LIKE, NOT LIKE
57 IS NULL, IS NOT NULL
58 IN
59 */
60 SELECT EMP_ID || EMP_NAME || SALARY "(설명)" /*특수문자 사용시 ""
61 FROM EMPLOYEE;
62
63 -- BETWEEN AND: 상한값과 하한값의 경계를 포함하는 값의 비교
64
65 /*
66 요구사항)
67 사원테이블에서 급여를 350만원보다 많이 받고 550만원보다 적게 받는 직원의 이름과 급여를
68 */
69
70 SELECT EMP_NAME, SALARY
71 FROM EMPLOYEE
72 WHERE SALARY BETWEEN 3500000 AND 5500000;
73
74 /*
75 요구사항)
76 사원테이블에서 이름이 김씨 성을 가진 직원의 이름과 급여를 조회하라
77 */
78
79 --LIKE : 패턴을 이용한 값의 비교시(%, _)
80 SELECT EMP_NAME, SALARY
81 FROM EMPLOYEE
82 WHERE EMP_NAME NOT LIKE '김%';
83
84 /*
85 요구사항) 부서가 없는 사원을 조회하라
86 */
87 SELECT *
88 FROM EMPLOYEE
89 WHERE DEPT_ID IS NOT NULL; /*NULL 컬럼 조건처리: IS NULL | IS NOT
90
91 /*
92 요구사항)
93 부서번호가 60번 이거나 90번인 사원의 이름, 부서코드, 급여를 조회하라
94 */
95
96 SELECT EMP_NAME, DEPT_ID, SALARY
97 FROM EMPLOYEE

```

<xe> 25-03-21

2025년 3월 27일 목요일 오후 4:37

```
98 WHERE DEPT_ID IN (60,90);
99
100 /*1. 춘 기술대학교의 학과 이름과 계열을 표시하시오. 단, 출력 헤더는 "학과 명", "계열"
101 으로 표시하도록 한다.*/
102
103 SELECT DEPARTMENT_NAME AS "학과 명" , CATEGORY AS "계열"
104 FROM TB_DEPARTMENT;
105
106 /*2. 학과의 학과 정원을 다음과 같은 형태로 화면에 출력한다. (다시 확인) */
107
108 SELECT DEPARTMENT_NAME || '의 정원은 ' || CAPACITY || ' 명 입니다.'
109 FROM TB_DEPARTMENT;
110
111
112 /*3. "국어국문학과" 에 다니는 여학생 중 현재 휴학중인 여학생을 찾아달라는 요청이
113 들어왔다. 누구인가? (국문학과와 '학과코드'는 학과 테이블 (TB_DEPARTMENT)을 조회해서
114 찾아 내도록 하자) */
115
116 SELECT DEPARTMENT_NO, DEPARTMENT_NAME
117 FROM TB_DEPARTMENT;
118
119 SELECT STUDENT_SSN
120 FROM TB_STUDENT;
121
122
123 SELECT STUDENT_NAME
124 FROM TB_STUDENT
125 WHERE DEPARTMENT_NO = '001' AND ABSENCE_YN = 'Y' AND
126
127 /*LIKE : 포함*/
128
129 /*4. 도서관에서 대출 도서 장기 연체자 들을 찾아 이름을 게시하고자 한다. 그 대상자들의
130 학번이 다음과 같을 때 대상자들을 찾는 SQL 구문을 작성하시오.
131 A513079, A513090, A513091, A513110, A513119*/
132
133 SELECT STUDENT_NAME
134 FROM TB_STUDENT
135 WHERE STUDENT_NO IN ('A513079', 'A513090', 'A513091',
136
137 /*5. 입학정원이 20 명 이상 30 명 이하인 학과들의 학과 이름과 계열을 출력하시오.*/
138
139 SELECT DEPARTMENT_NAME, CATEGORY
140 FROM TB_DEPARTMENT
141 WHERE CAPACITY BETWEEN 20 AND 30;
142
143
144
145 /*6. 춘 기술대학교는 총장을 제외하고 모든 교수들이 소속 학과를 가지고 있다. 그럼 춘
146 기술대학교 총장의 이름을 알아낼 수 있는 SQL 문장을 작성하시오.*/
```

```

147
148 SELECT PROFESSOR_NAME
149 FROM TB_PROFESSOR
150 WHERE DEPARTMENT_NO IS NULL;
151
152 /*7. 혹시 전산상의 착오로 학과가 지정되어 있지 않은 학생이 있는지 확인하고자 한다.
153 어떠한 SQL 문장을 사용하면 될 것인지 작성하시오.*/
154
155 SELECT STUDENT_NO, STUDENT_NAME
156 FROM TB_STUDENT
157 WHERE DEPARTMENT_NO IS NULL;
158
159 /*8. 수강신청을 하려고 한다. 선수과목 여부를 확인해야 하는데, 선수과목이 존재하는
160 과목들은 어떤 과목인지 과목번호를 조회해보시오.*/
161
162 SELECT CLASS_NO
163 FROM TB_CLASS
164 WHERE PREATTENDING_CLASS_NO IS NOT NULL;
165
166
167 /*9. 춘 대학에는 어떤 계열 (CATEGORY) 들이 있는지 조회해보시오.*/
168
169 SELECT DISTINCT CATEGORY
170 FROM TB_DEPARTMENT;
171
172
173
174 /*10. 02 학번 전주 거주자들의 모임을 만들려고 한다. 휴학한 사람들은 제외한 재학중인
175 학생들의 학번, 이름, 주민번호를 출력하는 구문을 작성하시오.*/
176
177 SELECT STUDENT_NO
178 FROM TB_STUDENT;
179
180 SELECT *
181 FROM TB_STUDENT;
182
183 SELECT
184 STUDENT_NO, STUDENT_NAME, STUDENT_SSN
185 FROM
186 TB_STUDENT
187 WHERE
188 ENTRANCE_DATE LIKE '02%' AND
189 ABSENCE_YN = 'N' AND
190 STUDENT_ADDRESS LIKE '%전주%';
191
192
193
194
195

```

196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214

200511 281 2721 7001 04 4 27